Stock Trend Prediction with Multi-Granularity Data: A Contrastive Learning Approach with Adaptive Fusion

Min Hou[1], Chang Xu[2], Yang Liu[1], Weiqing Liu[2], Jiang Bian[2]
Le Wu[3], Zhi Li[1], Enhong Chen[1], Tie-Yan Liu[2]
[1]School of Data Science, University of Science and Technology of China, Hefei, China
[2]Microsoft Research, Beijing, China; [3]Hefei University of Technology, Hefei, China
{minho, liuyang0, zhili03}@mail.ustc.edu.cn, cheneh@ustc.edu.cn,
{chanx, weiqing.liu, jiang.bian, tie-yan.liu}@microsoft.com, lewu@hfut.edu.cn

ABSTRACT

Stock trend prediction plays a crucial role in quantitative investing. Given the prediction task on a certain granularity (e.g., daily trend), a large portion of existing studies merely leverage market data of the same granularity (e.g., daily market data). In financial investment scenarios, however, there exist amounts of finer-grained information (e.g., high-frequency data) that contain more detailed investment signals beyond the original granularity data. This motivates us to investigate how to leverage multi-granularity market data to enhance the accuracy of stock trend prediction. Some straightforward methods, such as concatenating finer-grained data as features or fusing with a model based on finer-grained features, may not lead to more precise stock trend prediction due to some unique challenges. First, the inconsistency of granularity between the target trend and finer-grained data could substantially increase optimization difficulty, such as the relative sparsity of the target trend compared with higher dimensions of finer-grained features. Moreover, the continuously changing financial market state could result in varying efficacy of heterogeneous multi-granularity information, which consequently requires a dynamic approach for proper fusion among them. In this paper, we propose the Contrastive Multi-Granularity Learning Framework (CMLF) to address these challenges. Particularly, we first design two novel contrastive learning objectives at the pre-training stage to address the inconsistency issue by constructing additional self-supervised signals relying on the inherent character of stock data. We also design a gate mechanism based on market-aware technical indicators to fuse the multi-granularity features at each time step adaptively. Extensive experiments on three real-world datasets show significant improvements of our approach over the state-of-the-art baselines on stock trend prediction and profitability in real investing scenarios.

CCS CONCEPTS

• **Applied computing** → **Economics**; • **Computing methodologies** → **Artificial intelligence**.

CIKM '21, November 1–5, 2021, Virtual Event, Australia.

ACM ISBN 978-1-4503-8446-9/21/11…$15.00
https://doi.org/10.1145/3459637.3482483

KEYWORDS

Stock Trend Prediction; Contrastive Learning; Financial Investment

ACM Reference Format:
Min Hou[1], Chang Xu[2], Yang Liu[1], Weiqing Liu[2], Jiang Bian[2], Le Wu[3], Zhi Li[1], Enhong Chen[1], Tie-Yan Liu[2]. 2021. Stock Trend Prediction with Multi-Granularity Data: A Contrastive Learning Approach with Adaptive Fusion. In *Proceedings of the 30th ACM Int'l Conf. on Information and Knowledge Management (CIKM '21), November 1–5, 2021, Virtual Event, Australia.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3459637.3482483

1 INTRODUCTION

Stock trend prediction, which contributes to discovering latent trading patterns and seeking profit-maximization strategies in real-world stock markets, has attracted increasing attention in the quantitative finance area [4, 6, 14, 24, 46, 48]. A good estimation of stock trend can not only help investors with arbitrage, but also help individuals and the government analyze the markets environment and avoid risks. Experienced analysts with diverse strategies and purposes tend to prompt stock trend prediction under variant temporal granularities, such as minutely frequency, hourly frequency, daily frequency, weekly frequency. For example, analysts being prone to short-term high-yield yet high-risk are likely to focus on finer-granularity prediction, such as hourly or daily. On the contrary, those seeking long-term low-volatility returns may aim to coarser-granularity prediction, such as monthly or yearly.

Along this line, most existing methods [4, 19, 38] are single-granularity oriented methods, which are developed on a specific granularity of price-volume data, with the goal of predicting the stock trend labels at the specific level, e.g., daily data for daily trend prediction. However, high-frequency price-volume data is vital for making intelligent investment decisions, which usually provide complementary detailed investment signals that are not covered in the original granularity data, e.g., minute level data for daily trend prediction. In real-world scenarios, a mature analyst will thoroughly investigate the state of a stock market at various temporal granularities. A common phenomenon is that the fluctuation of intra-day prices with large volume may result from large shareholders pulling up/down prices, indicating a future reversal of market trends. Taking the daily-frequency trend prediction as an example, in addition to looking at the daily-frequency information, analysts also pay attention to the stocks' detailed minutely-frequency performance. Apparently, taking advantage of multi-granularity data yields great potential to improve stock trend prediction. Unfortunately, integrating multi-granularity information in stock trend prediction is still under-explored with great challenges.

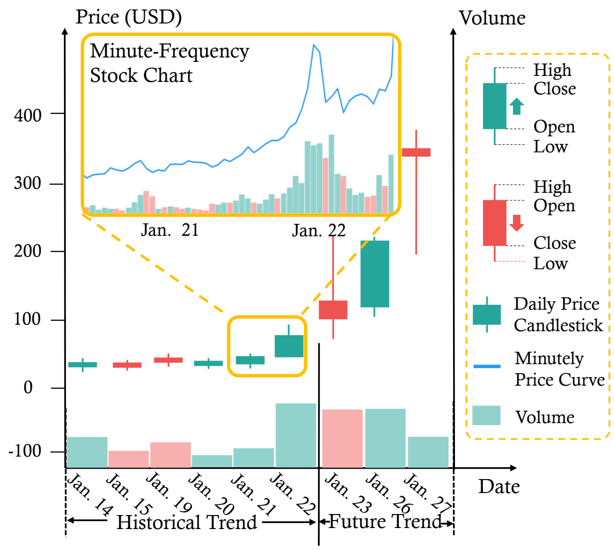


Figure 1: The daily and minute stock chart of GameStop during short squeeze event. The minute price pulling up on Jan.22 obviously implies the future stock trend.

On the one hand, how to extract information from multi-granularity data effectively is still an open issue. This issue comes from the inconsistency of granularity between data and label when using fine-grained data. There are dozens or hundreds of times more input signals carried by finer-grained data over the original granularity one while the training labels is relatively sparse [33]. For example, daily frequency data of a stock in one day may contain about 200 bits of information, and the corresponding minutely-frequency data contain about 50,000 bits. In contrast, the daily trend label contains much less information (32 bits for return ratio), which could make it hard to provide direct feedback signal to obtain informative multi-granularity embedding for stock trend prediction.

On the other hand, how to fuse multi-granularity information properly is not trivial. After obtaining the original and fine-grained embedding, an intuitive approach is to concatenate or add them directly. However, in real-world trading scenarios, the emphasis on different granularity features may vary. Therefore, regarding the fusion stage as a static process leads to performance inferior. Taking the GameStop short squeeze event [1] in NASDAQ as an example, in January 2021, a short squeeze of the stock of the American video game retailer GameStop and other securities took place, causing major financial consequences for certain hedge funds. The short squeeze caused the retailer's stock price to rise to over US $500 per share, almost 200 times its record low of $2.57. By observing and comparing high and low-frequency records on January 22, as shown in Figure 1, we find that GameStop's minute-frequency stock price experienced a sharp intraday increment, while the daily-frequency performance is relatively stable. If the model can accurately capture the price fluctuations and pay more attention to high-frequency data at this time, the dramatic pulling up of the price in the future trend would be possibly predicted.

[1] https://en.wikipedia.org/wiki/GameStop_short_squeeze

To address the above challenges, in this paper, we present an explorative study on stock trend prediction with a special focus on multi-granularity data. Then, we propose a Contrastive Multi-Granularity Learning Framework (CMLF) to exploit multi-granularity temporal information for the stock trend prediction task. Specifically, we first bridge the gap between our multi-granularity inputs and single-granularity target. We propose a novel contrastive learning mechanism with two objectives, i.e., cross-granularity and cross-temporal objectives. Next, we regard the fusion of multi-granularity data as a dynamic process and design a gate mechanism based on market-aware technical indicators to fuse the multi-granularity features at each time step adaptively. Finally, after a pre-training stage for our contrastive mechanisms, we optimize our proposed CMLF and generate trend prediction for the target stock.

In summary, the main contributions of this work include:

1) We present a focused study on multi-granularity data for stock trend prediction. To the best of our knowledge, this is among the first few studies to investigate how to adaptively fuse coarse and fine-grained data for stock trend prediction.
2) To address the task of stock trend prediction, we propose a novel Contrastive Multi-Granularity Learning Framework (CMLF) with two novel contrastive learning mechanisms to extract effective stock representations, and a specific gate mechanism to adaptively fuse multi-granularity data.
3) We conduct evaluation experiments on three real-world stock markets. The experiment results not only show significant improvements of our approach over current top systems, but also demonstrate the effectiveness of multi-granularity in the task of stock trend prediction.

2 PRELIMINARIES

In this section we introduce the contrastive learning mechanism used in this paper and describe the problem formulation of multi-granularity stock trend prediction.

2.1 Contrastive Learning

Contrastive learning is a promising class of self-supervised representation learning methods. The advantage of contrastive learning is to leverage the semantic dependency of data to construct training objectives. It can obtain feature representations that capture the essence of the data and improve data utilization efficiency.

The theoretical basis of contrastive learning is from the InfoMax [22] principle, which we instantiate here as maximizing the Mutual Information (MI) between two parts with semantic dependencies. Let $E_\psi : \mathcal{X} \to \mathcal{Y}$ defines a continuous and (almost everywhere) differentiable parametric function, where ψ represents a family of feature encoders (e.g., neural networks). $\mathcal{X}$ and $\mathcal{Y}$ denote the input and output domains. Different from traditional supervised learning methods that learn the encoder E_ψ from the supervision signals in labels, contrastive learning methods attempt to maximize the MI of the original input $\boldsymbol{x} \sim \mathcal{X}$ and the embedding $\boldsymbol{e} \sim E_\psi(\boldsymbol{x})$, which is defined as:

$$\mathrm{MI}(\boldsymbol{x}, \boldsymbol{e}) = \mathbb{E}_{(\boldsymbol{x},\boldsymbol{e})\sim p(\boldsymbol{x},\boldsymbol{e})}\left[\log \frac{p(\boldsymbol{x} \mid \boldsymbol{e})}{p(\boldsymbol{x})}\right]. \tag{1}$$

By maximizing MI, we can obtain the encoder that retains as much original information as possible. However, since the precise value of MI is difficult to compute, a common practise [33] is to utilize neural estimators to maximize the lower-bound of MI instead:

$$\mathrm{MI}(\boldsymbol{x}, \boldsymbol{e}) \geq \log(N) - \mathcal{L}_{\mathrm{N}}, \tag{2}$$

where $\mathcal{L}_{\mathrm{N}}$ is the contrastive loss function, which is defined as

$$\mathcal{L}_{\mathrm{N}} = \underset{(\boldsymbol{x},\boldsymbol{e})\sim p(\boldsymbol{x},\boldsymbol{e})}{\mathbb{E}} \left[-\log \frac{D_{\omega}(\boldsymbol{x}, \boldsymbol{e})}{D_{\omega}(\boldsymbol{x}, \boldsymbol{e}) + \sum_{i=1}^{N-1} D_{\omega}(\boldsymbol{x}_i', \boldsymbol{e})} \right]. \tag{3}$$

The lower bound is computed over a positive sample $(\boldsymbol{x}, \boldsymbol{e})$ from a joint distribution $p(\boldsymbol{x}, \boldsymbol{e})$ and $N-1$ negative samples $\{(\boldsymbol{x}_i', \boldsymbol{e})\}_{i=1}^{N-1}$ where $\boldsymbol{x}_i' \sim p(\boldsymbol{x})$. We define $D_{\omega} : \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}$ as a discriminator network. Both the encoder E_{ψ} and the discriminator D_{ω} are trained to jointly optimize Eqn.3. In fact, Eqn.3 is the categorical cross-entropy loss of classifying the positive sample correctly and is usually called InfoNCE loss [33]. Contrastive loss functions can also be in other forms, such as margin-based losses [39] and variants of NCE losses [16, 42]. Therefore, we can maximize MI by maximizing its lower bound, i.e. $\log(N) - \mathcal{L}_{\mathrm{N}}$, so that we can further acquire an encoder E_{ψ} that is capable of capturing significantly more information from the data itself.

2.2 Problem Formulation

Multi-granularity stock trend prediction takes historical multi-granularity features as input to predict the stock's future trend. This paper takes daily frequency as coarse-grained data and minute (15-minute) frequency as fine-grained data to illustrate our method. Actually, our framework is generally applicable to data of any arbitrary granularity. We take daily trend prediction as our prediction target, which is widely used in real-world investment scenarios.

Formally, the prediction model learns a function $\hat{y} = \mathcal{F}_{\Theta}\left(X^c, X^f\right)$, which maps the historical multi-granularity features to the stock trend label space. Specifically, $X^c = \left[x_1^c, \cdots, x_T^c\right] \in \mathbb{R}^{D\times T}$ represents the coarse-grained feature in the lag of past T time-steps. At each time-step t, x_t^c consists of D daily-frequency statistics, such as the highest price, opening price, lowest price, closing price, volume-weighted average price and trading volume. The fine-grained feature, i.e., minutely-frequency data, is denoted as $X^f = \left[x_1^f, \cdots, x_T^f\right] \in \mathbb{R}^{D\times K\times T}$, where each element x_t^f is composed of features from K equally divided time periods in a trading day, and each time-slot contains the same D statistics as the coarse-grained features. Given the coarse and fine-grained features, X^c and X^f, instead of predicting two classes (Rise or Fall), following [9], we aim to predict the return ratio of a stock which is formalized as $y = p_{T+2}/p_{T+1} - 1$, where p_t represents the volume-weighted average price of the stock at day t.

3 METHODOLOGY

This section elaborates on our Contrastive Multi-Granularity Learning Framework (CMLF) for stock trend prediction. First, we employ contrastive learning at the pre-training stage to handle the inconsistency of granularity between data and label. Relying on the inherent character of multi-granularity data, we construct additional self-supervised objectives to enhance the multi-granularity representation learning. Specifically, we design the cross-granularity and cross-temporal contrastive mechanisms in local and global scopes, which help the feature encoders to capture the coherence between coarse and fine-grained data, and the continuity of historical trend and current status. Moreover, to adapt to the dynamic and complicated stock market, we propose integrating the coarse and fine-grained features together by Adaptive RNN Cells. We first learn to capture the state of the market by designing technical indicators, and then feed them to the Adaptive Fusion Module to assist the model in deciding the emphasis over coarse or fine-grained features.

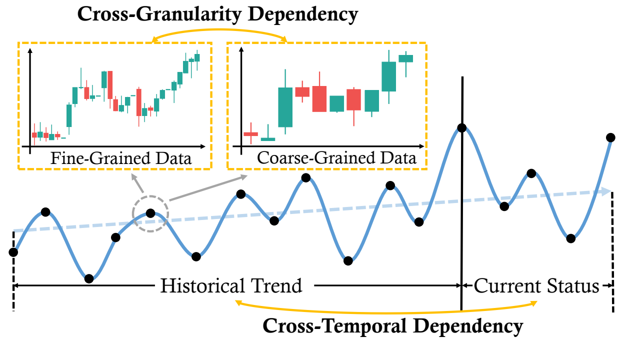


Figure 2: Intuition of the cross-granularity and cross-temporal dependency.

3.1 Pre-Training Stage: Contrastive Mechanisms

In the pre-training stage, we design two contrastive mechanisms to overcome the label sparsity caused by the inconsistent granularity between the feature and the target. That is, the gap between fine-grained features and coarse-grained targets, such as minute data for daily trend prediction. Specifically, we propose *Cross-Granularity Contrast* and *Cross-Temporal Contrast* in local and global scopes, respectively, to construct additional self-supervised training signals and learn effective multi-granularity features.

The intuition behind the Cross-Granularity Contrast Mechanism is to enable the feature encoders to capture semantic dependencies between coarse and fine-grained data. Since a pair of these coarse and fine-grained data of the same time period can be regarded as multi-perspective observations of one stock, we can hypothesize that there is latent coherency between them. For example, from Figure 2, we observe that the coarse and fine-grained data of a stock on a specific day have the same intra-day tendency of prices, though their granularity is different. Therefore, we propose capturing the underlying coherency between pair-wise coarse and fine-grained data by using contrastive learning techniques, which maximizes the MI between them.

Cross-Temporal Contrast Mechanism attempts to leverage the continuity of historical trends and current status of stock to construct additional training objectives. As shown in Figure 2, when we split a continuous stock price series into two parts from a certain time-step, the two parts should have a potential correlation. Since the whole trend of historical prices is upward, we can infer that the current situation is very likely to be consistent with the overall trend that is also upward. To preserve this correlation, we further propose making the current stock status and historical

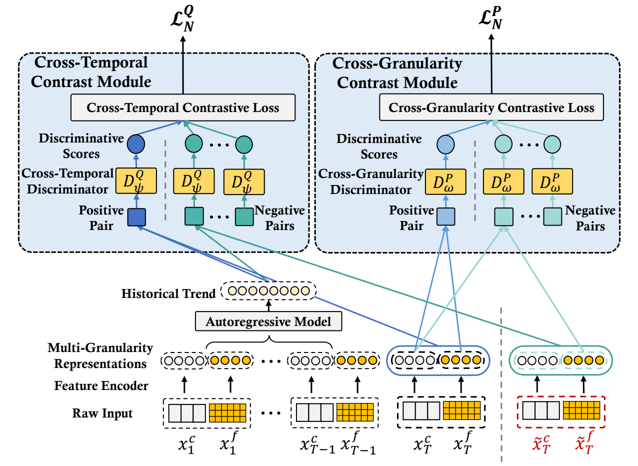


Figure 3: Cross-Granularity and Cross-Temporal Contrastive Learning Mechanisms at pre-training stage.

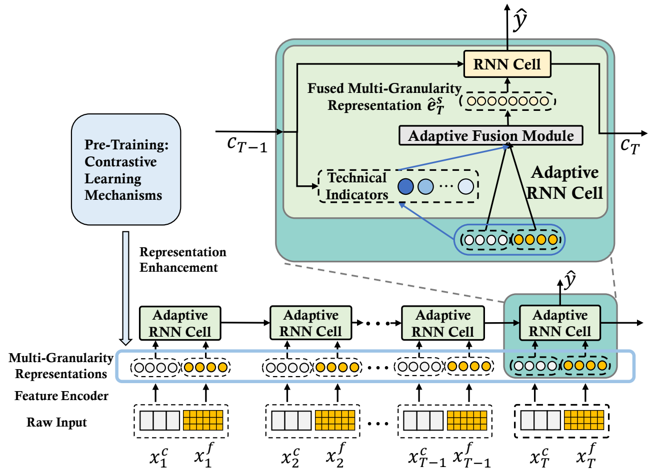


Figure 4: Adaptive Multi-Granularity Feature Fusion.

trend information of a stock sequence as similar as possible by maximizing their mutual information using a contrastive loss. The two mechanisms are introduced in detail below.

3.1.1 Cross-Granularity Contrast. We first design a local contrastive mechanism learning from coarse and fine-grained data pairs. As shown in Figure 3, at time-step t, the coarse-grained feature $\boldsymbol{x}_t^c$ and fine-grained feature $\boldsymbol{x}_t^f$ are fed into corresponding encoders, denoted as $En^c(\cdot)$ and $En^f(\cdot)$, respectively. The obtained representations are defined as $\boldsymbol{e}_t^c := En^c\left(\boldsymbol{x}_t^c\right)$ and $\boldsymbol{e}_t^f := En^f\left(\boldsymbol{x}_t^f\right)$. To enable the encoders to capture the semantic dependencies between coarse and fine-grained data, we propose maximizing the Mutual Information (MI) between the distribution of coarse-grained representation and fine-grained representation of the same stock on the same day, which is defined as

$$\mathrm{MI}(\boldsymbol{e}_t^f, \boldsymbol{e}_t^c) = \mathbb{E}_{(\boldsymbol{e}_t^f, \boldsymbol{e}_t^c) \sim \mathcal{P}_t^{f,c}} \log \frac{p(\boldsymbol{e}_t^f, \boldsymbol{e}_t^c)}{p(\boldsymbol{e}_t^f)p(\boldsymbol{e}_t^c)}, \tag{4}$$

where $\mathcal{P}_t^{f,c}$ represents the joint distribution of fine and coarse-grained representations, i.e., $(\boldsymbol{e}_t^f, \boldsymbol{e}_t^c) \sim \mathcal{P}_t^{f,c}$.

By maximizing the MI, the encoders can effectively extract shared information between coarse and fine-grained data. Since the precise value of MI is difficult to compute, we utilize neural estimators to maximize the lower-bound of MI instead [33]:

$$\mathrm{MI}\left(\boldsymbol{e}_t^f, \boldsymbol{e}_t^c\right) \geq \log(N) - \mathcal{L}_{\mathrm{N}}^{\mathrm{P}}, \tag{5}$$

where $\mathcal{L}_{\mathrm{N}}^{\mathrm{P}}$ is the cross-granularity contrastive loss. Then we optimize the contrastive loss $\mathcal{L}_{\mathrm{N}}^{\mathrm{P}}$, defined as:

$$\mathcal{L}_{\mathrm{N}}^{P} = - \mathop{\mathbb{E}}_{\mathcal{P}_t^{f,c}} \left[\log \frac{D_\omega^P\left(\boldsymbol{e}_t^f, \boldsymbol{e}_t^c\right)}{D_\omega^P\left(\boldsymbol{e}_t^f, \boldsymbol{e}_t^c\right) + \sum_{\tilde{\boldsymbol{e}}_t^f \in \hat{\mathrm{E}}_t^f} D_\omega^P\left(\tilde{\boldsymbol{e}}_t^f, \boldsymbol{e}_t^c\right)} \right], \tag{6}$$

where $D_\omega^P(\cdot,\cdot)$ is the *cross-granularity discriminator* that parameterized by ω. The negative sample $\tilde{\boldsymbol{e}}_t^f$ is randomly sampled from the marginal distribution of fine-grained data from other samples within one mini-batch, forming a set of $N-1$ elements, denoted by $\hat{\mathrm{E}}_t^f$.

We define cross-granularity discriminator D_ω^P to measure the similarity between a pair of representations. A simple log-bilinear model is used in our work, although other forms producing positive real scores can also work. The *cross-granularity discriminative score* of a pair of fine and coarse-grained representations is computed as

$$D_\omega^P\left(\boldsymbol{e}_t^f, \boldsymbol{e}_t^c\right) = \exp\left(\boldsymbol{e}_t^{f\top} \boldsymbol{W}^P \boldsymbol{e}_t^c\right), \tag{7}$$

where $\boldsymbol{W}^P$ is a learnable linear transformation matrix.

The principle behind the cross-granularity contrastive loss is to make $(\boldsymbol{e}_t^f, \boldsymbol{e}_t^c)$, i.e. the representations of positive pairs sampled from the joint distribution, as similar as possible, and make the negative pairs constructed by replacing $\boldsymbol{e}_t^f$ with randomly sampled $\tilde{\boldsymbol{e}}_t^f$ from the marginal distribution as dissimilar as possible. In this way, we can maximize MI between fine and coarse-grained representations by minimizing $\mathcal{L}_{\mathrm{N}}^{\mathrm{P}}$, so that we can further acquire encoders $En^c(\cdot)$ and $En^f(\cdot)$ which are capable of capturing the underlying coherence information from multi-granularity data.

3.1.2 Cross-Temporal Contrast. We propose a global contrastive mechanism which learns from historical trends and the current status. The multi-granularity representations extracted by encoders are denoted as $\boldsymbol{e}_t = [\boldsymbol{e}_t^c, \boldsymbol{e}_t^f]$. To extract the historical trend information buried in previous $t-1$ time-steps, we apply an autoregressive model $AR(\cdot)$ to summarize all $\boldsymbol{e}_{<t} = [\boldsymbol{e}_1, \boldsymbol{e}_2, \cdots, \boldsymbol{e}_{t-1}]$ in the latent space and produce a trend latent representation $AR(\boldsymbol{e}_{<t})$. To capture the continuity of historical trend and current status, we propose maximizing the MI between $AR(\boldsymbol{e}_{<t})$ and $\boldsymbol{e}_t$:

$$\mathrm{MI}(\boldsymbol{e}_t, AR(\boldsymbol{e}_{<t})) = \mathbb{E}_{\boldsymbol{e}_{\leq t} \sim \mathcal{P}_t^e} \log \frac{p(\boldsymbol{e}_t, AR(\boldsymbol{e}_{<t}))}{p(\boldsymbol{e}_t)p(AR(\boldsymbol{e}_{<t}))}, \tag{8}$$

where $\mathcal{P}_t^e$ defines the joint distribution of historical multi-granularity representations, i.e. $(\boldsymbol{e}_1, \boldsymbol{e}_2, \cdots, \boldsymbol{e}_t) \sim \mathcal{P}_t^e$.

Similar to solving the cross-granularity contrast, we also use neural estimators to maximize the lower-bound of MI. The corresponding cross-temporal contrast loss is defined as:

$$\mathcal{L}_{\mathrm{N}}^{Q} = -\mathop{\mathbb{E}}_{\mathcal{P}_t^e}\left[\log \frac{D_{\psi}^{Q}\left(\boldsymbol{e}_t, AR(\boldsymbol{e}_{<t})\right)}{D_{\psi}^{Q}\left(\boldsymbol{e}_t, AR(\boldsymbol{e}_{<t})\right) + \sum_{\tilde{\boldsymbol{e}}_t \in \hat{\mathrm{E}}_t} D_{\psi}^{Q}\left(\tilde{\boldsymbol{e}}_t, AR(\boldsymbol{e}_{<t})\right)}\right], \quad (9)$$

where $D_{\psi}^{Q}(\cdot,\cdot)$ is the *cross-temporal discriminator* parameterized by ψ. The negative sample $\tilde{\boldsymbol{e}}_t$ is randomly sampled from the marginal distribution of multi-granularity representations from other samples within one mini-batch, forming a set of $N-1$ elements denoted by $\hat{\mathrm{E}}_t$.

We define the cross-temporal discriminator D_{ψ}^{Q} in the form of log-bilinear:

$$D_{\psi}^{Q}\left(\boldsymbol{e}_t, AR\left(\boldsymbol{e}_{<t}\right)\right) = \exp(\boldsymbol{e}_t^{\top} \boldsymbol{W}^{Q} AR\left(\boldsymbol{e}_{<t}\right)). \quad (10)$$

By contrastive learning between positive and negative sample pairs, the cross-temporal contrast mechanism helps the encoders capture the stock inherent trend information.

3.2 Adaptive Multi-Granularity Feature Fusion

As the influence of multi-granularity information varies at different time-steps, we propose regarding the emphasis on coarse or fine-grained data as a dynamic process. We suppose that the importance of different granularity data is influenced by the market state. Therefore, we introduce technical indicators [27] to describe the market state and further design a gate-based multi-granularity fusion module accordingly to decide how to integrate coarse and fine-grained data at each time-step adaptively.

3.2.1 Technical Indicator Construction. We introduce technical indicators to describe the market state, which are widely used for providing reliable trading signals in technical analysis. The choice of technical indicators can be flexible. In this paper, we choose some traditional technical indicators as well as design two novel indicators based on the relationship between multi-granularity data inspired by a couple of intuitions from the real-world.

On the one hand, we follow previous works [14, 19] and utilize some well-recognized technical indicators that are mathematically calculated on a time series of prices and returns. We denote the set of them as **I**. The indicators fall into three categories, (1) those on volatility (e.g., the upper and lower bands of *Bollinger Bands* [2]), (2) those on momentum (e.g., *Money Flow Index* [3]), and (3) those on trend (e.g., *Moving Average Convergence Divergence*[4]).

On the other hand, we construct two novel MI indicators by using the discriminators learned from the pre-training stage to describe the relationship between coarse- and fine-grained data. Empirically, we find two factors may influence the investors' attention to the coarse and fine-grained information. The first one is the MI of the coarse and fine-grained data at the current time-step, which can imply the short-term volatility of prices. For instance, large MI indicates that the fine-grained data fluctuate widely and may contain rich detailed information that needs to be paid more attention. The second one is the MI between historical trend and the present status, which reflects the long-term coherence through the continuity between historical and present prices. Therefore, we use the discriminators learned at the pre-training stage to construct two novel MI related indicators accordingly: cross-granularity contrast indicator and cross-temporal contrast indicator.

Remind that Eqn. (6) is the categorical cross-entropy loss of classifying the positive sample correctly, with $\frac{D_{\omega}^{P}}{\sum_{\mathrm{M}} D_{\omega}^{P}}$ being the prediction model where the set $\mathrm{M} = \{\boldsymbol{e}_t^{f}\} \cup \hat{\mathrm{E}}_t^{f}$ contains 1 positive sample and $N-1$ negative samples. Let us rewrite the optimal probability for $\mathcal{L}_{\mathrm{N}}^{P}$ as $p\left(d = i \mid \mathrm{M}, \boldsymbol{e}_t^{c}\right)$ with $[d = i]$ representing sample i is the 'positive' sample $\boldsymbol{e}_t^{f,(i)}$. The other $d \neq i$ represents the negative sample $\boldsymbol{e}_t^{f,(d)} \in \hat{\mathrm{E}}_t^{f}$. The probability of sample d being positive can be derived as follows:

$$\begin{aligned} p(d = i \mid \mathrm{M}, \boldsymbol{e}_t^{c}) &= \frac{p(\boldsymbol{e}_t^{f,(i)}, \boldsymbol{e}_t^{c}) \prod_{k \neq i} p(\boldsymbol{e}_t^{f,(k)})}{\sum_{j=1}^{N} p(\boldsymbol{e}_t^{f,(j)}, \boldsymbol{e}_t^{c}) \prod_{k \neq j} p(\boldsymbol{e}_t^{f,(k)})} \\ &= \frac{\frac{p(\boldsymbol{e}_t^{f,(i)}, \boldsymbol{e}_t^{c})}{p(\boldsymbol{e}_t^{f,(i)}) p(\boldsymbol{e}_t^{c})}}{\sum_{j=1}^{N} \frac{p(\boldsymbol{e}_t^{f,(j)}, \boldsymbol{e}_t^{c})}{p(\boldsymbol{e}_t^{f,(j)}) p(\boldsymbol{e}_t^{c})}}. \end{aligned} \quad (11)$$

Therefore, we conclude that:

$$D_{\omega}^{P*}\left(\boldsymbol{x}_t^{f}, \boldsymbol{x}_t^{c}\right) \propto \frac{p\left(\boldsymbol{e}_t^{f}, \boldsymbol{e}_t^{c}\right)}{p\left(\boldsymbol{e}_t^{f}\right) p\left(\boldsymbol{e}_t^{c}\right)}, \quad (12)$$

$$D_{\psi}^{Q*}\left(\boldsymbol{e}_t, AR\left(\boldsymbol{e}_{<t}\right)\right) \propto \frac{p\left(\boldsymbol{e}_t, AR\left(\boldsymbol{e}_{<t}\right)\right)}{p\left(\boldsymbol{e}_t\right) p(AR\left(\boldsymbol{e}_{<t}\right))}. \quad (13)$$

Since the *Pointwise Mutual Information* (PMI) of a pair of samples is defined as: $\mathrm{PMI}(x, y) = \log \frac{p(x,y)}{p(x)p(y)}$, where $x \in X$ and $y \in Y$. Consequently, the discriminative scores computed by the optimal discriminators D_{ω}^{P*} and D_{Ω}^{Q*} are:

$$\alpha_t = D_{\omega}^{P*}\left(\boldsymbol{e}_t^{f}, \boldsymbol{e}_t^{c}\right), \quad (14)$$

$$\beta_t = D_{\psi}^{Q*}\left(\boldsymbol{e}_t, AR\left(\boldsymbol{e}_{<t}\right)\right). \quad (15)$$

α_t and β_t can be regarded as the indicators of the PMI between *coarse & fine-grained data* and *the historical trend & present status.*

3.2.2 Adaptive Multi-Granularity Fusion Module. The indicator of cross-granularity contrast describes the short-term volatility of prices by quantifying the consistency of fine and coarse-grained data; and the indicator of cross-temporal contrast describes the long-term volatility through the consistency between historical and present prices. We leverage these two indicators as well as traditional technical indicators to learn how to adaptively fuse the information of different granularity at each time step. Formally, the state of Adaptive RNN Cell at time t is

$$\boldsymbol{c}_t = RNN(\boldsymbol{c}_{t-1}, \hat{\boldsymbol{e}}_t), \quad (16)$$

where $\hat{\boldsymbol{e}}_t$ is the adaptively fused multi-granularity features. The prediction result $\hat{y}$ is generated from the matrix transformation on the hidden state of the last time-step $\boldsymbol{c}_T$. The proposed adaptive multi-granularity fusion module is defined as:

[2]https://en.wikipedia.org/wiki/Bollinger_Bands
[3]https://en.wikipedia.org/wiki/Money_flow_index
[4]https://en.wikipedia.org/wiki/MACD

Table 1: Details of dataset splitting

Property	Datasets		
	CSI300	CSI800	NASDAQ100
Training Period	2007/02/16-2014/12/31		2005/01/01-2013/12/31
Validation Period	2015/01/01-2016/12/31		2014/01/01-2015/12/31
Test Period	2017/01/01-2020/01/01		2016/01/01-2020/06/01
# Total Stocks	749	1,687	171
# Total Records	908,606	2,412,678	559,586
# Features	6	6	5

$$
\begin{aligned}
g_f &= \sigma\left(\boldsymbol{W}_c^1 \boldsymbol{e}_t^c + \boldsymbol{W}_f^1 \boldsymbol{e}_t^f + \sum_{\rho \in \mathbf{I} \cup \{\alpha,\beta\}} \boldsymbol{W}_\rho^1 \rho_t + b^1\right) \\
g_c &= \sigma\left(\boldsymbol{W}_c^2 \boldsymbol{e}_t^c + \boldsymbol{W}_f^2 \boldsymbol{e}_t^f + \sum_{\rho \in \mathbf{I} \cup \{\alpha,\beta\}} \boldsymbol{W}_\rho^2 \rho_t + b^2\right) \\
\hat{\boldsymbol{e}}_t &= g_f \circ \boldsymbol{e}_t^f + g_c \circ \boldsymbol{e}_t^c,
\end{aligned} \tag{17}
$$

where $\boldsymbol{g}_f$ and $\boldsymbol{g}_c$ are the fine and coarse-grained gates. Other notations, including $\boldsymbol{W}_c^1, \boldsymbol{W}_f^1, \boldsymbol{W}_i^1, b^1$ and $\boldsymbol{W}_c^2, \boldsymbol{W}_f^2, \boldsymbol{W}_i^2, b^2$, are learnable parameters. $\mathbf{I}$ is a set of traditional technical indicators, in this work, including upper band of Bollinger Bands, lower band of Bollinger Bands, Money Flow Index, and Moving Average Convergence Divergence. σ denotes the sigmoid activation function and $\circ$ is the element-wise product. By using the cross-granularity and cross-temporal indicators to construct fine and coarse-grained gates, the multi-granularity features are fused adaptively.

3.3 Model Learning Strategy

The training process of the entire model consists of two stages. During the pre-training stage, we minimize an integrated loss including supervised stock trend prediction loss, two contrastive loss, and L_2 regularization:

$$
\mathcal{L}_1 = \sum_{s=1}^{S} \|y^s - \hat{y}^s\|^2 + \lambda_1 \sum_{s=1}^{S}\sum_{t=1}^{T} \mathcal{L}_{\mathrm{N}}^P + \lambda_2 \sum_{s=1}^{S}\sum_{t=1}^{T} \mathcal{L}_{\mathrm{N}}^Q + \frac{\lambda_3}{2}\|\Theta\|_F^2, \tag{18}
$$

where $\lambda_1, \lambda_2, \lambda_3$ are the hyper-parameters to balance different losses. S is the total number of stock records.

After that, we fix the encoders En^f, En^c and the contrastive discriminators D_ω^{P*}, D_ψ^{Q*} obtained from the previous stage, and learn the other parameters from scratch. The loss function is

$$
\mathcal{L}_2 = \sum_{s=1}^{S} \|y^s - \hat{y}^s\|^2 + \frac{\lambda}{2}\|\Theta\|_F^2, \tag{19}
$$

where the λ is the hyper-parameter of the L_2 regularization.

4 EXPERIMENTS

In this section, we verify the effectiveness of our framework on the real-world stock markets. We focus on making daily-frequency stock trend prediction based on both daily-frequency and minutely-frequency data as input in our experiments. Actually, our framework is generally applicable to data of any arbitrary granularity. We report the main prediction results on three real-world datasets and analyze the profits in real financial investing scenarios to verify the effectiveness of our method. We also conduct extensive analytical experiments, including visualization of the enhanced stock representations, verifying the effectiveness of MI indicators and case studies.

4.1 Experiment Setups

4.1.1 Data Collection. We evaluate our models on real-world stock data. We collect stock sequences from Qlib[5], an AI-oriented quantitative investment platform. Our dataset consists of low-frequency (daily) and high-frequency (15-min) price-volume data over constituent stocks from three major stock indices over the world: CSI300, CSI800 and NASDAQ100. We split the sequences by time, forming training sets, validation sets and test sets. The details of the three datasets are listed in Table 1. There are six commonly used statistics extracted as features for CSI300 and CSI800 datasets, including the highest price, the opening price, the lowest price, the closing price, volume-weighted average price, and trading volume. NASDAQ dataset contains the remaining five features except the volume-weighted average price. The data are adjusted for dividends and splits, and normalized by the Z-Score method.

4.1.2 Evaluation Metrics. As described in Section 2.2, we take the multi-granularity stock trend prediction as a regression problem. Hence, the most straightforward evaluation metrics, including Root Mean Squared Error **(RMSE)** and Mean Absolute Error **(MAE)** are used in our paper.

4.1.3 Comparison Methods. We compare our method with competitive baselines, which can be categorize into four groups.

- The first group consists of classical time series forecasting models, including **Linear Regression (LR)** and **Transformer** [34].
- The second group consists of current top systems for stock trend prediction based on daily data. **SFM** [48] aims to capture trading patterns from investors with different trading modes inspired by Fourier Transform. **ALSTM** [29] contains a temporal attentive aggregation layer based on normal LSTM. **Adv-ALSTM** [12] is a variant of ALSTM with adversarial training, which is claimed to be a state-of-the-art method using daily-frequency data for daily trend prediction.
- The third group contains varients of our model using different granularities of data. **Coarse-Grained RNN** and **Fine-Grained RNN** use only coarse-grained data or fine-grained data, respectively. The input of **Multi-Grained RNN** is the concatenation of two granularities of data. The basic architectures of these methods are consistent with CMLF.
- The fourth group contains four ablation counterparts of our Contrastive Multi-Granularity Learning Framework **CMLF**. **CMLF w/o AMFM** stands for CMLF without the Adaptive Multi-granularity Fusion Module. **CMLF w/o CG** and **CMLF w/o CT** represent CMLF without the enhancement of cross-granularity contrast or cross-temporal contrast. For CMLF w/o CG, at the pre-training stage, we remove $\mathcal{L}_{\mathrm{N}}^P$ in Eqn. (18) and set α_t to zero in Eqn. (17) at the fine-tuning stage. Similarly, we can get CMLF w/o CT. **CMLF w/o MII**

[5]https://github.com/microsoft/qlib

Table 2: Performance of stock trend prediction on CSI300, CSI800 and NASDAQ100.

Method	CSI300		CSI800		NASDAQ100	
	RMSE↓	MAE↓	RMSE↓	MAE↓	RMSE↓	MAE↓
LR	4.2761	4.0382	3.2507	2.9174	4.9414	2.2029
Transformer	3.6961	3.6304	2.1877	2.1265	3.2589	2.5626
SFM	2.8251	2.6348	2.0907	1.5572	2.9177	2.1245
ALSTM	2.8251	2.4941	2.5761	2.1836	3.3174	2.6223
ADV-ALSTM	3.1770	2.8754	2.3462	1.8253	3.3201	2.6233
Coarse-Grained RNN	2.3429	2.1634	1.8821	1.5470	3.4083	2.1281
Fine-Grained RNN	2.2680	2.1067	1.7839	1.4923	3.2982	2.5850
Multi-Grained RNN	2.0784	1.8209	1.6797	1.3213	2.4443	1.4853
CMLF w/o CT	1.3408	1.1396	1.3947	1.0688	2.4644	1.5140
CMLF w/o CG	1.3810	1.1001	1.7277	1.3723	2.5169	1.5837
CMLF w/o AMFM	0.8300	0.6573	1.4599	1.0596	2.4589	1.4935
CMLF w/o MII	0.7533	0.5865	1.4236	1.0428	2.4375	1.4622
CMLF	**0.6975**	**0.5275**	**1.3865**	**0.9922**	**2.4345**	**1.4596**

represents CMLF without the two proposed MI Indicators in the Adaptive Multi-granularity Fusion Module.

4.1.4 Implementation Details. We use a fully-connected layer as En^c and a 2-layer GRU to encode En^f. For the autoregressive model, we use another 2-layer GRU. We set the hidden size to 64. In the fine-tuning stage, we feed $\hat{\boldsymbol{e}}_t$ to a 2-layers GRU model to get the final prediction result $\hat{y}$. The traditional technical indicators are generated using the TA-Lib[6] library. For all the methods, we optimize them by mini-batch Adam until convergence and tune the hyper-parameters via grid search on the validation set with the learning rate selected from $[10^{-4}, 10^{-3}, 10^{-2}]$. The coefficient of L_2 regularization is tuned amongst $[10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}]$. For a fair comparison, RNN backbones in ALSTM and Adv-ALSTM are searched from the traditional RNN and its variants LSTM and GRU, and the number of layers is chosen from [1, 2]. The Transformer we compared has 2 encoder layers, and the number of heads is searched from [2, 5]. We tune the hyper-parameters of the baseline methods both from the values listed in their source code and similar range as used for the proposed method, and we report the best performance. The coefficients of two contrastive learning objectives (λ_1 and λ_2) in our proposed methods are tuned from $[0.1, 0.2, \cdots, 1.0]$. After conducting grid-searching, the optimal λ_1 and λ_2 are set to [1.0, 1.0] for CSI300 and [0.2, 1.0] for CSI800, NASDAQ100. Other hyper-parameters are empirically set as follows: the embedding size is set to 64, the input window size T is 20, the batch size is 300 for CSI300, 800 for CSI800, and 200 for NASDAQ100. We repeat each experiment 20 times and report the average results. All experiments are conducted on a Linux server with two RTX 2080 Ti GPUs.

4.2 Experiment Results

4.2.1 Performance Comparison. Table 2 shows the prediction performance of all methods on three datasets. We predict the stock trend of all individual stocks in the datasets and report the average performance. We make the following observations: Fine-Grained RNN has no significant improvement over Coarse-Grained RNN, indicating that high dimensional input is hard to model with limited coarse-grained labels. Besides, Multi-Granularity RNN achieves better performance than all single-granularity models, which verifies that combining the multi-granularity features can boost the performance. Furthermore, CMLF significantly outperforms the Multi-granularity RNN, demonstrating that our adaptive method can integrate the features better than baselines.

We also observe that both CMLF w/o CT and CMLF w/o CG surpass Multi-Grained RNN, which has the same architecture but no additional contrastive learning objectives. These results demonstrate the effectiveness of contrastive mechanisms. Moreover, CMLF achieves better performance than CMLF w/o CG and CMLF w/o CT, indicating that the two contrastive mechanisms can mutually enhance each other. Furthermore, CMLF shows superior performance over CMLF w/o AMFM, verifying the adaptive module's critical effects in fusing different granularities of data. The large improvement of CMLF over CMLF w/o MII illustrates the effectiveness of the two proposed MI indicators.

Overall, CMLF outperforms all the baselines to a large margin, including the existing stock trend prediction methods on all three datasets and two evaluation metrics. The results justify contrastive learning mechanisms' effectiveness and the adaptive feature fusion approach for the stock trend prediction task.

4.2.2 Market Trading Simulation. To further verify the profitability of our method in real financial investing scenarios, we conduct a trading simulation. We employ the *TopK-Drop Strategy* [4], which is a straightforward but popular trading strategy to conduct back-testing on real stock data. Initially, investors invest in the *Top K* stocks with the highest predicted ranking score. On each trading day, the *Drop* number of held stocks with the worst prediction score will be sold, and the same number of unheld stocks with the best prediction score will be bought. We conduct back-testing on real stock data with considering practical constraints (i.e., with 0.15% opening fee, 0.25% closing fee and 9.5% price limit threshold). Note

[6]https://ta-lib.org/

Table 3: Quantitative risk analysis of trading strategies in the market simulation.

Method	RankIC↑	RankIR↑	AR (%)↑	Sharpe↑	MDD↑
Benchmark	-	-	9.0848	0.5072	-0.3705
LR	0.0427	0.2341	-3.6710	-0.1756	-0.4293
Transformer	0.0686	0.4813	2.6587	0.1365	-0.4339
SFM	0.0837	0.6620	10.9742	0.5592	-0.3702
ALSTM	0.0820	0.6443	9.6631	0.4947	-0.3773
ADV-ALSTM	0.0867	0.6709	16.3034	0.8673	-0.2512
Coarse-Grained RNN	0.0809	0.6450	14.7204	0.7604	-0.2928
Fine-Grained RNN	0.0812	0.6462	16.2142	0.8330	-0.3295
Multi-Grained RNN	0.0857	0.6472	18.1177	0.9098	-0.3031
CMLF	**0.0948**	**0.7239**	**19.5480**	**1.0181**	**-0.2502**

that *Topk-Drop* algorithm always sells *Drop* stocks every trading day, which guarantees a fixed turnover rate.

The *Benchmark* denotes the CSI300 index itself, which can reflect the overall performance of the market. Note that benchmark denotes the profit of CSI300 index itself, which is not the prediction of any approach. So the benchmark has no Rank IC and Rank IR. Each model is initialized with \$ 1,000,000 for trading simulation. Figure 5 compares the cumulative profit curves corresponding to different approaches as well as the benchmark. From this figure, we find that CMLF achieves the highest yield over almost the entire testing period despite the market volatility. At the end of the test period, after deducting practical constraints, the benchmark (black line) yield is 24.24%. The CMLF (red line) yield is 67.10%, indicating that the trading strategy based on CMLF can achieve an excess return rate of nearly 42.86%.

Moreover, we conduct further quantitative analysis on trading strategy as shown in Table. 3. Five important evaluation metrics are adopted to evaluate the model performance, including rank correlation (*RankIC* [19] and *RankIR* [13]), risk-adjusted return (*Sharpe Ratio* [31]), return indicator (*Annualized Return, AR*), and risk indicator (*Max Drawdown, MDD* [26]). CMLF achieves the best performance over all these quantitative indicators, which indicates that our model can make a steady profit. For instance, CMLF achieves the best *Sharpe Ratio* result 1.0181, while the *Sharpe Ratio* of the benchmark is only 0.5072. It further demonstrates that CMLF has relatively higher returns and lower risks.

4.2.3 Visualization of Stock Representations. To illustrate the effectiveness of our proposed contrastive learning mechanisms for stock representations, we dig into a case on the learned stock embeddings. We randomly select one trading day (Jan. 9$^{\text{th}}$, 2017) in the test set of CSI300, then apply t-SNE [25] to visualize the representation of the stock with the top 20% and bottom 20% return ratio of this day. Figure 6 (a) and (b) present the 2D embeddings visualization for Multi-grained RNN and CMLF, respectively. As shown in the figure, the clustering boundary of CMLF is more clear than that of Multi-granularity RNN. This indicates that CMLF with two additional contrastive learning objectives can learn higher-quality features, which can effectively differentiate stocks of different trends. Therefore, CMLF can capture rich information in multi-granularity stock data and reach better performance.

4.2.4 Effectiveness of MI indicators and case studies. In order to verify the effectiveness of the two MI indicators, further experiments are designed to investigate whether the indicators are capable of describing the state changes of the market. We expect to observe the the pattern of the changes of indicators when severe market fluctuations happen. Intuitively, once the fluctuations take place, the performance of coarse and fine-grained data ought to be different, and the current state will also diverge from historical trends. To prove this, two adjacent days (10/31/2019 and 11/1/2019) are randomly selected from the testset of CSI300. Then, stock collections with large price fluctuations ($y_{t-1} \in [-\delta, \delta]$ and $y_t \in [y|y < -\epsilon \cup y > \epsilon]$, here we set $\delta = 1, \epsilon = 2$), are extracted. Through statistical analysis, it is discovered that stock records with the two indicator values decreasing occupied 87.5% of the collections. This shows that the cross-granularity indicator precisely detects the inconsistency of multi-granularity data, while the cross-temporal indicator unearth the incontinuity of the historical trend and current status accurately.

We conduct a case study to analyze how our method effectively fuses multi-granularity features adaptively to fit the changes in the stock market. We test the previously showed case, the short squeeze event of GameStop, by our model. Figure 1 shows the minute and daily frequency stock prices of the test sample. We can observe that the price fluctuates seriously on Jan. 22, and there is a dramatic pulling up of the price on Jan.23. Along with the stock market changes, our MI indicators capture the fluctuation successfully. Specifically, the cross-granularity MI indicator changes from 9.53 to 3.77, and the cross-temporal MI indicator decrease from 2.88 to 0.46, implying there is a severe fluctuation of stock prices. Our model precisely predicts the pulling up on Jan.23.

5 RELATED WORK

5.1 Stock Trend Prediction

Enormous efforts have been spent studying how to make accurate stock trend predictions so as to seek maximized profits. Although the Efficient Market Hypothesis (EMH) [11] and the random walk theory imply that the stock price is unpredictable [28, 36] in a totally efficient market, most of stock markets in the real world, fortunately, are not completely efficient [2].

Among all efforts that have been devoted to predicting stock price trends, to be specific, the price change rates or whether the

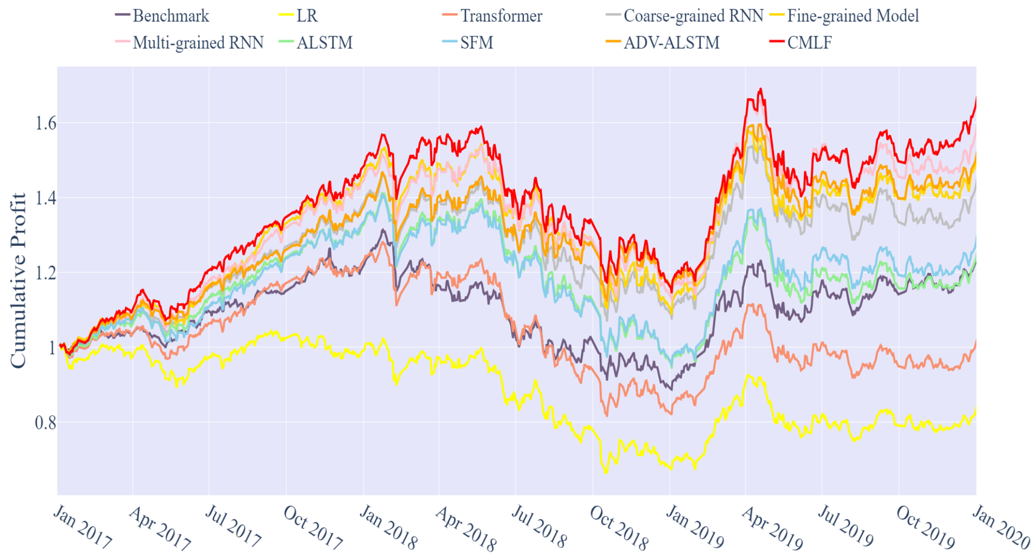


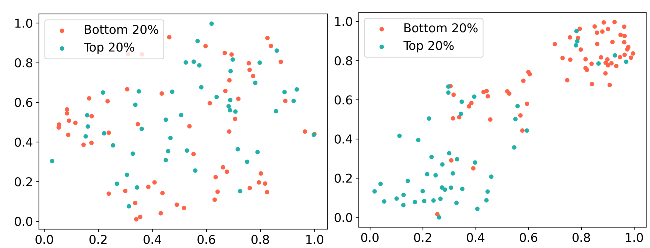
Figure 5: Cumulative profit with TopK-Drop Trading Strategy.

price will go up or down [1, 45], one of the major branches attempts to rely on quantitative technical information [10] to construct effective stock prediction models. Stock trend prediction methods mainly fall under two categories, Fundamental Analysis (FA) and Technical Analysis (TA) [10]. FA methods are developed with the explosion of finance alternative data, such as news [8, 24], social media [32, 40] and company's announcements [44]. On the other hand, TA methods extract price-volume information from historical trading data and use machine learning algorithms for prediction. Most previous works [4, 12, 18, 19, 21, 37, 38] straightforwardly use the historical price-volume features of the identical frequency/granularity as the prediction target. Although there exist a few works [17, 20, 41, 43, 47] using "multi-scale" information for time series analysis, they usually use two types of scale information, i.e., wavelet-based [48] and down-sampling based [23]. We clarify that these types of multi-scale information are different from what we refer to by "multi-granularity", i.e., different statistical scales.

5.2 Contrastive Learning

Contrastive learning techniques are a promising class of self-supervised representation learning methods. The main idea is to train an encoder to be contrastive between representations that captures statistical or semantic dependencies of interest and those that do not. A contrastive approach usually employs a scoring function, training the encoder to increase the score on positive or related samples, e.g. co-occurring words in a sentence or different data argumentation results of the same image, and decrease the scores of negative pairs generated from unrelated and corrupted data.

The motivation of contrastive learning is the InfoMax [22] principle, which we here instantiate as maximizing the Mutual Information (MI) between two parts with semantic dependencies. For example, Deep InfoMax [16] maximizes MI between local and global representation using MINE [3]. Contrastive Prediction Coding [33] assumes an order in the features extracted from the original data and used summary features to predict future features. SimSLR [5] extractes views and generates positive samples using different augmentations of data points. Nowadays, contrastive learning has been successfully applied in image [5, 15], speech [33] and graph [35] data. While most contrastive learning studies focus on learning representations from unlabeled data, we use it as the regularization term to improve supervised tasks following some recent works [7, 30].



(a) Multi-Granularity RNN (b) CMLF

Figure 6: Visualization of stock representations produced by (a) Multi-granularity RNN and (b) CMLF on CSI300 dataset.

6 CONCLUSION AND FUTURE WORK

In this paper, we study the problem of fusing multi-granularity data for stock trend prediction. To the best of our knowledge, we are among the first few studies to address this problem. We propose a novel multi-granularity learning method named Contrastive Multi-granularity Learning Framework, which fuses coarse and fine-grained information for stock trend prediction. In the future, we plan to explore the impact of more than two granularities of data on prediction models and extend our work to the field of more general time series analysis tasks.

REFERENCES

[1] Ayodele Ariyo Adebiyi, Aderemi Oluyinka Adewumi, and Charles Korede Ayo. 2014. Comparison of ARIMA and artificial neural networks models for stock price prediction. Journal of Applied Mathematics 2014 (2014).
[2] Nicholas Barberis and Richard Thaler. 2003. A survey of behavioral finance. Handbook of the Economics of Finance 1 (2003), 1053–1128.
[3] Mohamed Ishmael Belghazi, Aristide Baratin, Sai Rajeshwar, Sherjil Ozair, Yoshua Bengio, Aaron Courville, and Devon Hjelm. 2018. Mutual information neural estimation. In International Conference on Machine Learning. 531–540.
[4] Chi Chen, Li Zhao, Jiang Bian, Chunxiao Xing, and Tie-Yan Liu. 2019. Investment behaviors can tell what inside: Exploring stock intrinsic properties for stock trend prediction. In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2376–2384.
[5] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. A simple framework for contrastive learning of visual representations. ICML (2020).
[6] Yingmei Chen, Zhongyu Wei, and Xuanjing Huang. 2018. Incorporating corporation relationship via graph convolutional neural networks for stock price prediction. In Proceedings of the 27th ACM International Conference on Information and Knowledge Management. 1655–1658.
[7] Yu-An Chung and James Glass. 2020. Improved speech representations with multi-target autoregressive predictive coding. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics). 2353-–2358.
[8] Shumin Deng, Ningyu Zhang, Wen Zhang, Jiaoyan Chen, Jeff Z Pan, and Huajun Chen. 2019. Knowledge-driven stock trend prediction and explanation via temporal convolutional network. In Companion Proceedings of The 2019 World Wide Web Conference. 678–685.
[9] Yi Ding, Weiqing Liu, Jiang Bian, Daoqiang Zhang, and Tie-Yan Liu. 2018. Investor-imitator: A framework for trading knowledge extraction. In Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 1310–1319.
[10] Robert D Edwards, WHC Bassetti, and John Magee. 2007. Technical analysis of stock trends. CRC press.
[11] Eugene F Fama. 1970. Efficient capital markets: A review of theory and empirical work. The journal of Finance 25, 2 (1970), 383–417.
[12] Fuli Feng, Huimin Chen, Xiangnan He, Ji Ding, Maosong Sun, and Tat-Seng Chua. 2019. Enhancing Stock Movement Prediction with Adversarial Training. IJCAI (2019).
[13] Thomas H Goodwin. 1998. The information ratio. Financial Analysts Journal 54, 4 (1998), 34–43.
[14] Yuechun Gu, Da Yan, Sibo Yan, and Zhe Jiang. 2020. Price Forecast with High-Frequency Finance Data: An Autoregressive Recurrent Neural Network Model with Technical Indicators. In Proceedings of the 29th ACM International Conference on Information & Knowledge Management. 2485–2492.
[15] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. 2020. Momentum contrast for unsupervised visual representation learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 9729–9738.
[16] R Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Phil Bachman, Adam Trischler, and Yoshua Bengio. 2019. Learning deep representations by mutual information estimation and maximization. In International Conference on Learning Representations.
[17] Min Hou, Le Wu, Enhong Chen, Zhi Li, Vincent W. Zheng, and Qi Liu. 2019. Explainable Fashion Recommendation: A Semantic Attribute Region Guided Approach. In Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19. 4681–4688.
[18] Chang Li, Dongjin Song, and Dacheng Tao. 2019. Multi-task recurrent neural networks and higher-order Markov random fields for stock price movement prediction: Multi-task RNN and higer-order MRFs for stock price classification. In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 1141–1151.
[19] Zhige Li, Derek Yang, Li Zhao, Jiang Bian, Tao Qin, and Tie-Yan Liu. 2019. Individualized indicator for all: Stock-wise technical indicator optimization with stock embedding. In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 894–902.
[20] Zhi Li, Hongke Zhao, Qi Liu, Zhenya Huang, Tao Mei, and Enhong Chen. 2018. Learning from history and present: Next-item recommendation via discriminatively exploiting user behaviors. In Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 1734–1743.
[21] Tao Lin, Tian Guo, and Karl Aberer. 2017. Hybrid Neural Networks for Learning the Trend in Time Series. In Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17. 2273–2279.
[22] Ralph Linsker. 1988. Self-organization in a perceptual network. Computer 21, 3 (1988), 105–117.
[23] Guang Liu, Yuzhao Mao, Qi Sun, Hailong Huang, Weiguo Gao, Xuan Li, Jianping Shen, Ruifan Li, and Xiaojie Wang. 2020. Multi-scale Two-way Deep Neural Network for Stock Trend Prediction. In Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence,IJCAI-20. 4555–4561.
[24] Qikai Liu, Xiang Cheng, Sen Su, and Shuguang Zhu. 2018. Hierarchical complementary attention network for predicting stock price movements with news. In Proceedings of the 27th ACM International Conference on Information and Knowledge Management. 1603–1606.
[25] Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. Journal of machine learning research 9, Nov (2008), 2579–2605.
[26] Malik Magdon-Ismail and Amir F Atiya. 2004. Maximum drawdown. Risk Magazine 17, 10 (2004), 99–102.
[27] John J Murphy. 1999. Technical analysis of the financial markets: A comprehensive guide to trading methods and applications. Penguin (1999).
[28] Thien Hai Nguyen, Kiyoaki Shirai, and Julien Velcin. 2015. Sentiment analysis on social media for stock movement prediction. Expert Systems with Applications 42, 24 (2015), 9603–9611.
[29] Yao Qin, Dongjin Song, Haifeng Chen, Wei Cheng, Guofei Jiang, and Garrison Cottrell. 2017. A dual-stage attention-based recurrent neural network for time series prediction. IJCAI (2017), 2627–2633.
[30] Mirco Ravanelli and Yoshua Bengio. 2019. Learning speaker representations with mutual information. InterSpeech (2019).
[31] William F Sharpe. 1994. The sharpe ratio. Journal of portfolio management 21, 1 (1994), 49–58.
[32] Jianfeng Si, Arjun Mukherjee, Bing Liu, Qing Li, Huayi Li, and Xiaotie Deng. 2013. Exploiting topic based twitter sentiment for stock prediction. In Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). 24–29.
[33] Aäron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation Learning with Contrastive Predictive Coding. ArXiv abs/1807.03748 (2018).
[34] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. arXiv preprint arXiv:1706.03762 (2017).
[35] Petar Veličković, William Fedus, William L. Hamilton, Pietro Liò, Yoshua Bengio, and R Devon Hjelm. 2019. Deep Graph Infomax. In International Conference on Learning Representations. https://openreview.net/forum?id=rklz9iAcKQ
[36] Steven Walczak. 2001. An empirical analysis of data requirements for financial forecasting with neural networks. Journal of management information systems 17, 4 (2001), 203–222.
[37] Jingyuan Wang, Ze Wang, Jianfeng Li, and Junjie Wu. 2018. Multilevel wavelet decomposition network for interpretable time series analysis. In Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2437–2446.
[38] Jingyuan Wang, Yang Zhang, Ke Tang, Junjie Wu, and Zhang Xiong. 2019. AlphaStock: A Buying-Winners-and-Selling-Losers Investment Strategy using Interpretable Deep Reinforcement Attention Networks. In Proceedings of the 25th ACM SIGKDD. 1900–1908.
[39] Xiaolong Wang and Abhinav Gupta. 2015. Unsupervised learning of visual representations using videos. In Proceedings of the IEEE international conference on computer vision. 2794–2802.
[40] Huizhe Wu, Wei Zhang, Weiwei Shen, and Jun Wang. 2018. Hybrid deep sequential modeling for social text-driven stock prediction. In Proceedings of the 27th ACM International Conference on Information and Knowledge Management. 1627–1630.
[41] Likang Wu, Zhi Li, Hongke Zhao, Zhen Pan, Qi Liu, and Enhong Chen. 2020. Estimating early fundraising performance of innovations via graph-based market environment model. In Proceedings of the AAAI, Vol. 34. 6396–6403.
[42] Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. 2018. Unsupervised feature learning via non-parametric instance discrimination. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 3733–3742.
[43] Chang Xu, Weiran Huang, Hongwei Wang, Gang Wang, and Tie-Yan Liu. 2019. Modeling local dependence in natural language with multi-channel recurrent neural networks. In Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 33. 5525–5532.
[44] Wentao Xu, Weiqing Liu, Chang Xu, Jiang Bian, Jian Yin, and Tie-Yan Liu. 2021. REST: Relational Event-Driven Stock Trend Forecasting. In Proceedings of the Web Conference 2021 (Ljubljana, Slovenia) (WWW '21). New York, USA, 1–10.
[45] Yumo Xu and Shay B Cohen. 2018. Stock movement prediction from tweets and historical prices. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). 1970–1979.
[46] Yiying Yang, Zhongyu Wei, Qin Chen, and Libo Wu. 2019. Using External Knowledge for Financial Event Prediction Based on Graph Neural Networks. In Proceedings of the 28th ACM International Conference on Information and Knowledge Management. 2161–2164.
[47] Kai Zhang, Hao Qian, Qing Cui, Qi Liu, Longfei Li, Jun Zhou, Jianhui Ma, and Enhong Chen. 2021. Multi-Interactive Attention Network for Fine-grained Feature Learning in CTR Prediction. In Proceedings of the 14th ACM International Conference on Web Search and Data Mining. 984–992.
[48] Liheng Zhang, Charu Aggarwal, and Guo-Jun Qi. 2017. Stock price prediction via discovering multi-frequency trading patterns. In Proceedings of the 23rd ACM SIGKDD. 2141–2149.